1

National Defence / Défense nationale

RATE COMPATIBLE PUNCTURED CODES, THEIR USE IN MILITARY SATELLITE COMMUNICATION SYSTEMS

by

Lyle C. Wagner

DEFENCE RESEARCH ESTABLISHMENT OTTAWA
TECHNICAL NOTE 89-19

Canada

August 1989
Ottawa

National Defence Défense nationale

RATE COMPATIBLE PUNCTURED CODES, THEIR USE IN MILITARY SATELLITE COMMUNICATION SYSTEMS

by

Lyle C. Wagner
EHF SATCOM Section
Electronics Division

DEFENCE RESEARCH ESTABLISHMENT OTTAWA
TECHNICAL NOTE 89-19

PCN
041LM

August 1989
Ottawa

ABSTRACT

The problem of interfacing older military terminal equipments, which work exclusively from an internal timing source, into a communication system that requires all terminals to synchronise to the network clock is addressed in this paper. These data communication systems would typically operate with a fixed symbol transmission rate and employ end-to-end convolutional coding for channel errors. A scheme is proposed which would allow a clock discrepancy of up to ±4% between the terminal clock and the network clock. The method proposed is to add control bits to the information stream between terminals. The control bits would allow the effective information rate to either increase or decrease by ±4%. The overall symbol rate of the communication link is held constant by selecting a higher rate convolutional code from the set of rate compatible punctured codes (RCPCs). This paper describes the methodology used to select an optimal RCPC and presents some examples. The probability of error for a selected code is compared with that of a convolutional code for an application typical of a satellite communication system. The conclusion is that the increase in signal to noise required to sustain the same probability of error is minimal.

RÉSUMÉ

Le problème d'interface entre de vieux terminaux militaires, fonctionnant exclusivement à l'aide d'une horloge interne, et un système de communication, requérant la synchronisation de tout terminal avec l'horloge du réseau, est présenté dans cet article. Ces systèmes de communication de donneés opèrent typiquement à un taux fixe de transmission de symboles et emploient un codage convolutionel à chaque extrémité pour les erreurs de transmission. Un arrangement permettant un écart maximal de ±4% entre l'horloge du réseau et celle du terminal est proposé. La méthode suggéréé consiste à ajouter des bits de contrôle au début d'information entre les terminaux. Ces bits de contrôle permettraient une augumentation ou une diminution de 4% du taux effectif de transmission d'information. Le taux global de transmission de symbole de la liaison en communication est maintenu constant en sélectionnant un code convolutionnel à taux plus élevé à partir d'un ensemble de codes perforés ayant un taux compatible. Cet article décrit la méthodologie utilisée pour sélectionner un code perforé optimum et présente quelques exemples. La probabilité d'erreur d'un code choisi est comparée à celle d'un code convolutionnel pour une application typique d'un système de communication par satellite. L'article conclut que l'augmentation du rapport signal-bruit, requise pour obtenir la même probabilité d'erreur, est minimale.

EXECUTIVE SUMMARY

Many modern Military Satellite Communication (MILSATCOM) Systems assume that user equipment will be able to synchronise to a network clock. However, this is often not feasible with older military equipments which have no input for an external clock. To solve the possible rate discrepancy the system interface to the equipment must either speed up or slow down the overall end-to-end data rate. This paper proposes a solution to this problem for data communication systems with fixed symbol transmission rates and convolutional coding for channel errors.

The usual forward error correction (FEC) code specified in the MILSATCOM systems is a rate 1/2 convolutional code. If a slightly higher rate code could be found (i.e. 13/24), extra overhead bits could be added to the data stream without impacting the coded data rate. The overhead bits would be used to indicate that either more than normal data is being sent or less then normal data is being sent, and to carry the additional data.

This paper uses a rate 1/2 convolutional code and then punctures it by deleting parity check symbols to obtain a rate 13/24, rate compatible punctured code (RCPC). This code permits thé inclusion of 2 overhead bits for every 24 data bits transmitted. The puncturing method chosen is optimised for minimum impact on error rate. Using error bounding and simulation, the RCPC code is compared to the convolutional code for a typical MILSATCOM application. The result is that to support a probability of error of 10^{-5}, the signal to noise ratio required for the RCPC is 0.3 dB above the convolutional code. The scheme using the RCPC code will support a clock discrepancy of ±4% between the data source and the network.

CONTENTS

1. INTRODUCTION

Many existing military communications equipment do not support external clocking, but work exclusively from an internal timing source. If the equipment is connected to a network and its internal clock is not exactly matched to the network clock, then the interface will pass too many or too few data bits per second. The end result will be a communication link that regularly loses information bits and synchronisation. Any data communication system that must support this type of interface must rationalise the inconsistency in clock rates. Unfortunately many networks assume that all connected equipments accept an external clocking source or have an internal clock that is matched exactly to the network clock. This document proposes the use of a rate compatible punctured convolutional (RCPC) code in the data link standard of the communication systems to support equipment that works exclusively from an internal clocking source.

The data interface problem is described in more detail in Section 2 along with a general concept for correcting the problem using a variable rate code. Section 3 defines what a RCPC code is and the specific code parameters used in this document. Section 4 outlines how the optimum puncturing table is derived. The simulation of a communications system model used to verify the theory is presented in Section 5. Section 6 gives the results. Enhancements to the scheme to counteract jamming are given in Section 7 and other uses of RCPC are proposed in Section 8. Section 9 presents the conclusions of the paper.

2. STATEMENT OF PROBLEM

Many data communication systems operate with a fixed symbol transmission rate and end-to-end forward error correction (FEC) for channel errors. These systems typically assume that the user equipment would be able to synchronise to the network clock in the interface equipment. However, this is not possible with many military equipments which have no input for an external clock. To solve this problem the interface equipment must either speed up or slow down the effective end-to-end information rate. The use of an elastic buffer will only provide a temporary solution to the problem and eventually an overflow or underflow will occur (depending on whether the input rate is too fast or too slow).

The FEC code typically used in MILSATCOM equipment is a rate 1/2 convolutional code. If a slightly higher rate code could be found (e.g. 13/24), extra overhead bits could be added to the data stream without changing the coded data rate. The overhead bits would carry any data overflow and would be used to indicate more information is being sent or less information is being sent than normal.

For consideration in this paper, data is assumed to be normally blocked into groups of 24 bits and then encoded into 48 bits with a rate 1/2 encoder. If by using the higher code rate two extra bits were added, the network would then be able to support a variation of ± 1 bit in 24, or a user clock rate discrepancy of $\pm 4\%$. In order for the overall coded data rate to remain constant, a rate 26/48 (or 13/24) encoder will be required. Table 1 describes one method of using the two extra bits to identify the number of data bits in the block and carry the additional data bit when it occurs. If a constant 24

bits of data were being transmitted, the logic would alternate between sending 23 bits and 25 bits.

BIT LOCATION					RESULT
1	2	3	...	26	
0	unused		data		23 data bits
1	data				25 data bits

Table 1. Definition of overhead bits in data link standard.

3. DEFINITION OF RCPC CODES

The use of convolutional codes usually means that the coding rate is fixed to simple ratios (1/2, 1/3, 2/3). Non-simple ratios result in complicated encoders and decoders which are difficult to implement; however another option exists in the form of punctured convolutional codes. The concept of punctured codes was first introduced by Cain, Clark, and Geist [1] to obtain a simpler Viterbi decoding scheme for rate 2/3 and rate 3/4 codes. This concept was then expanded by Hagenauer [2] to make the coding scheme fit the transmission rate. The general concept of a RCPC code is to puncture a low rate 1/N code with period P to obtain a family of codes with rate $P/(P+\zeta)$, where ζ can be varied from 1 to $(N-1)P$. From this family of codes, a code is derived with the appropriate code rate. To obtain the rate 13/24 code required for the problem defined in this paper the following parameters were used

$N = 2$, initial code rate = 1/2
$P = 13$, ζ can be varied from 1 to 13
$\zeta = 11$

A punctured code can be implemented with a scheme similar to that depicted in Fig. 1. The generator matrix would be the same as for any convolutional encoder and would operate as per normal. The puncturing would be accomplished through the use of a puncturing table. The output from the encoder would be compared with the value in the puncturing table. A zero in the table means that the code symbol is not transmitted, a one in the table means that the code is transmitted. In the example, the 4^{th} bit of the upper branch is not transmitted and the 2^{nd} bit of the lower branch is not transmitted.

The decoding of the coded information is relatively straightforward. The critical factor is that the encoder and decoder must be in perfect bit synchronisation. The decoder would use a standard Viterbi algorithm set to decode the data coded with the corresponding encoding matrix. In addition, the decoder would have the same puncturing table as the transmitter. A one in the table indicates that the encoded bit would have been transmitted and the decoder can calculate the maximum likelihood matrix as normal. A zero in the the table indicates that the encoded bit would have been been deleted and the decoder would not be able to use this bit in the calculation of the maximum likelihood matrix.

4. DETERMINATION OF THE OPTIMAL PUNCTURING TABLE

As stated in [2], there is no known constructive method for determining the optimum combination of generator matrix and puncturing table for a RCPC code family. Instead the usual method of determination is through a computer search, as was done in this report.

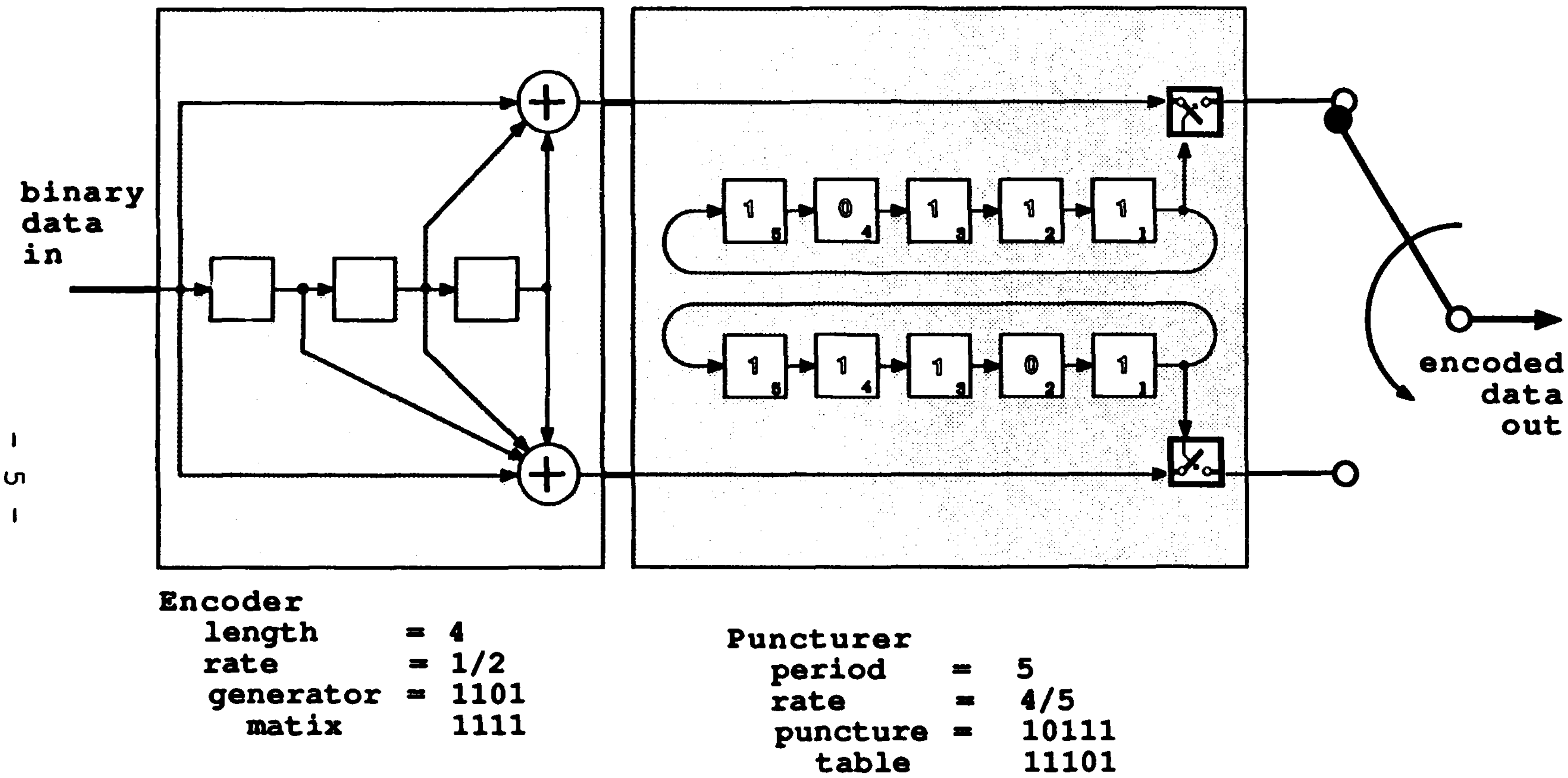


Fig. 1. Simplified diagram of encoder and code puncturer.

The following lists the basic steps used to determine the optimum puncturing table.

1. Select a suitable generator matrix. In this report a rate 1/2 code, constraint length 7, with maximum free distance was chosen.

2. For the value of ζ selected ($\zeta = 11$ in this report) construct a puncturing table.

3. Calculate the bound on the expected number of bit errors for the generator matrix and puncturing table selected. The method to calculate the upper bound was first proposed by Viterbi and Omura [3] and is given in Appendix A.

4. Repeat steps 2 and 3 for all possible constructions of the puncturing table.

5. Chose the puncturing table construct which yields the best bit error rate performance (lowest upper bound).

Using the method described above the following parameters where determined for the *RCPC code*.

	binary	octal
Generator matrix	1111001	171
	1011011	133
Puncture table	0111110111111	07677
	1111111111111	17777

5. SIMULATION

The capabilities of a RCPC code in a binary symmetric channel (BSC) with additive white Gaussian noise (AWGN) was determined through a Monte Carlo computer simulation. This result was then compared with the calculated upper bound. The communications system model for this simulation is given in Fig. 2. The model is divided into three major sections, transmitter, communication link, and receiver. These section are described in the following paragraphs.

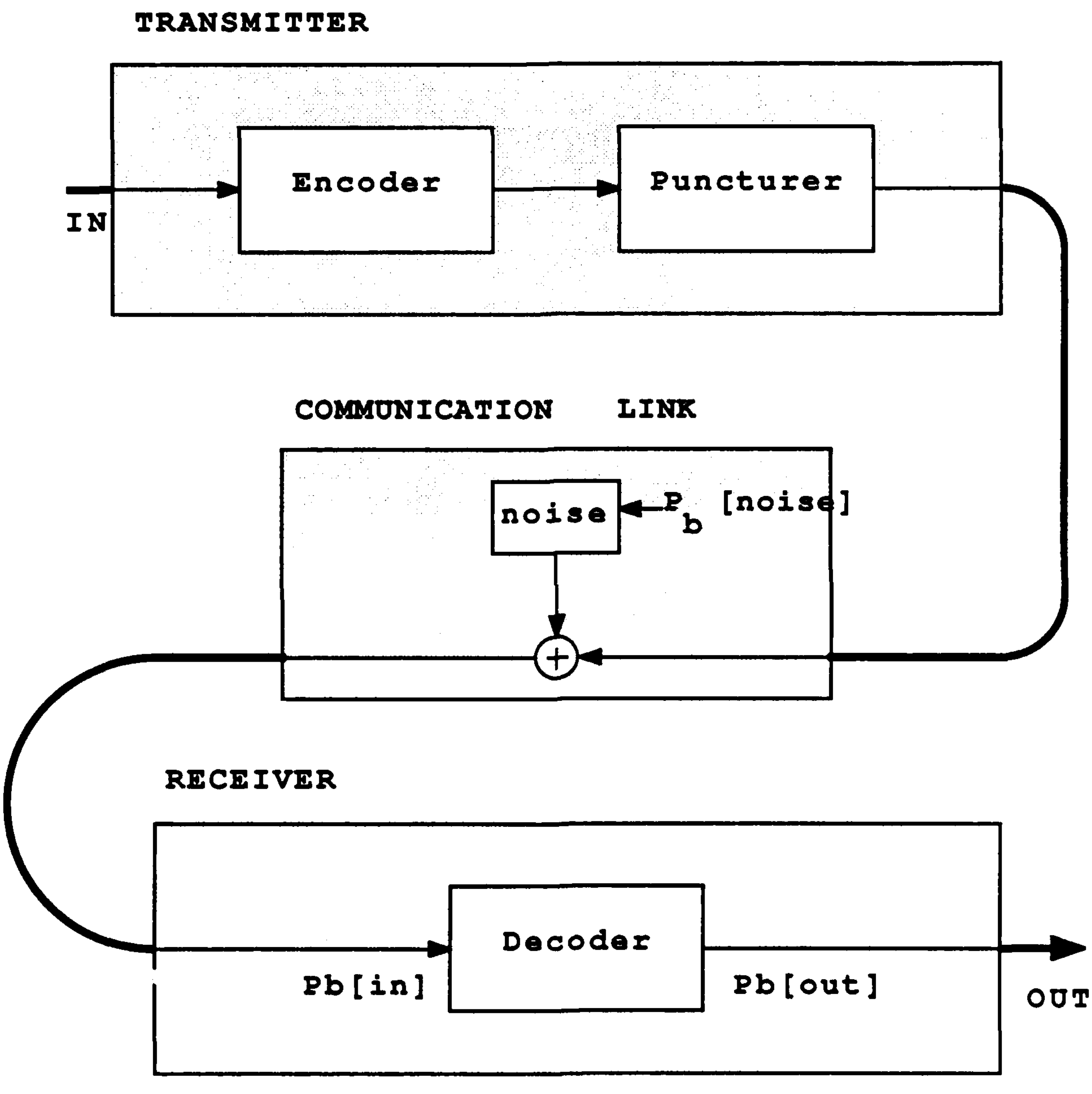


Fig. 2. Simplified block diagram of communication model.

Transmitter

The transmitter is modelled as a random binary source, a convolutional encoder, and code puncturer. Random digit data is obtained by rounding to the nearest binary integer the output of a random number generator. The number generator is of the multiplicative congruential type which generates numbers uniformly distributed between 0 and 1. The encoder is a rate one-half encoder with a constraint length and generator determined by the user. The code puncturer is as shown in Fig. 1, with the period and puncturing table determined by the user.

Communication Link

The communication link is modelled as a binary bit stream with AWGN causing individual bits to be corrupted. The determination of whether to corrupt a bit is determined by the theoretical probability of bit error due to noise (P_b[noise]), which is an input to the simulation. A separate random number is created by the random number generator described above. If the random number is less than or equal to P_b[noise], the value of the bit is inverted (0 changed to 1 or 1 changed to 0). If the random number is greater than P_b[noise], the value of the bit is unchanged.

Receiver

The receiver is modelled as a hard-decision maximum-likelihood Viterbi decoder that has been modified to decode RCPC codes. The decoded data is stored for later analysis.

6. RESULTS

The simulation of the communication system model was used to determine the accuracy of the bound on the probability of bit error. Using the same generator matrix and puncturing table as were calculated in section 3, simulations were run to generate output bit error rates (BER) for input P_b of 10^{-2} to 10^{-6}. In addition, simulations were run for an unpunctured code of the same generator matrix. These results are shown in Fig. 3. For the range of interest, and specifically a BER of 10^{-5}, the upper bound and the simulation results are relatively close.

The results shown in Fig. 3 also demonstrates the performance degradation caused by using the punctured code instead of the unpunctured code. The input P_b level required to produce a BER of 10^{-5} are given in Table 2.

Code Type	Input P_b for an output BER = 10^{-5}	
	Upper Bound	Simulation
RCPC code	0.0133	0.0148
Convolutional code	0.0171	0.0192

Table 2. Input P_b required for an output BER of 10^{-5}.

17-APR-89 13:51:58

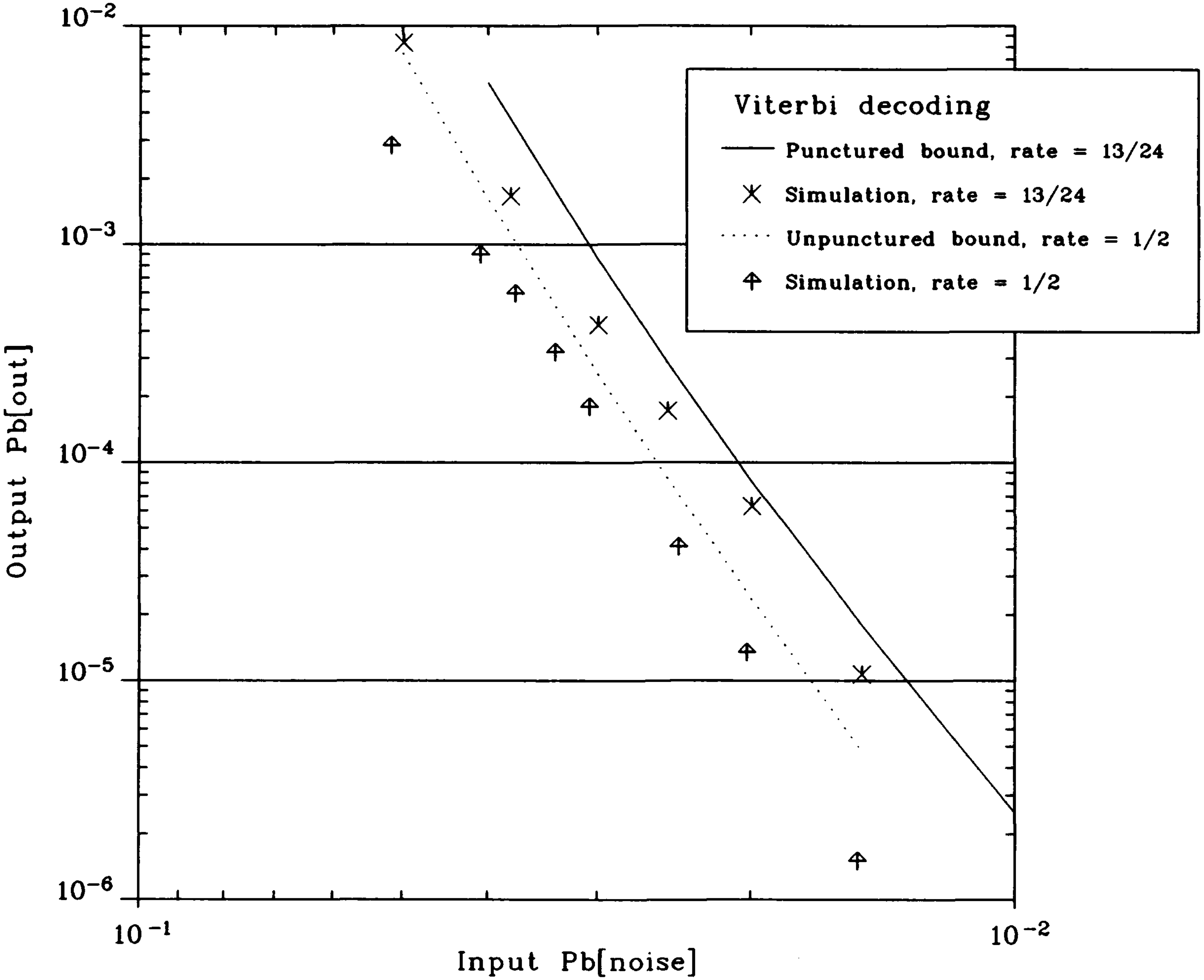


Fig 3. Comparison of simulation and bounds

To quantify the comparison of RCPC codes to standard convolution codes the modulation scheme must be taken into consideration. The modulation scheme used in this paper is M-ary frequency shift keying (MFSK), where M=8. From Proakis [4], the probability of symbol error (P_s) is given by

$$P_s = \sum_{n=1}^{M-1} (-1)^{n+1} \binom{M-1}{n} \frac{1}{n+1} e^{-nk\gamma_b/(n+1)} \qquad \ldots (1)$$

where γ_b = signal to noise ratio (SNR) per bit
k = 3, number of bits per symbol.

The bit error probability (P_b) is obtained from P_s by

$$P_b = \frac{2^{k-1}}{2^k - 1} P_s \qquad \ldots (2)$$

By combining equations (1) and (2) we get P_b expressed as a function of γ_b, designated as

$$P_b = f[\gamma_b] \qquad \ldots (3)$$

The value of γ_b, given P_b, is determined using Newton's iteration method from equation (3).

The upper bounds for the RCPC code and the equivalent convolution code are shown in Fig. 4 versus the signal to noise ratio for MFSK. For reference, the uncoded case is also shown in Fig. 4. The degradation in using the RCPC code over the normal convolution code is 0.3 dB. The advantage gained from using a RCPC code is that the link is now able to handle a variable rate.

17-APR-89 13:52:27

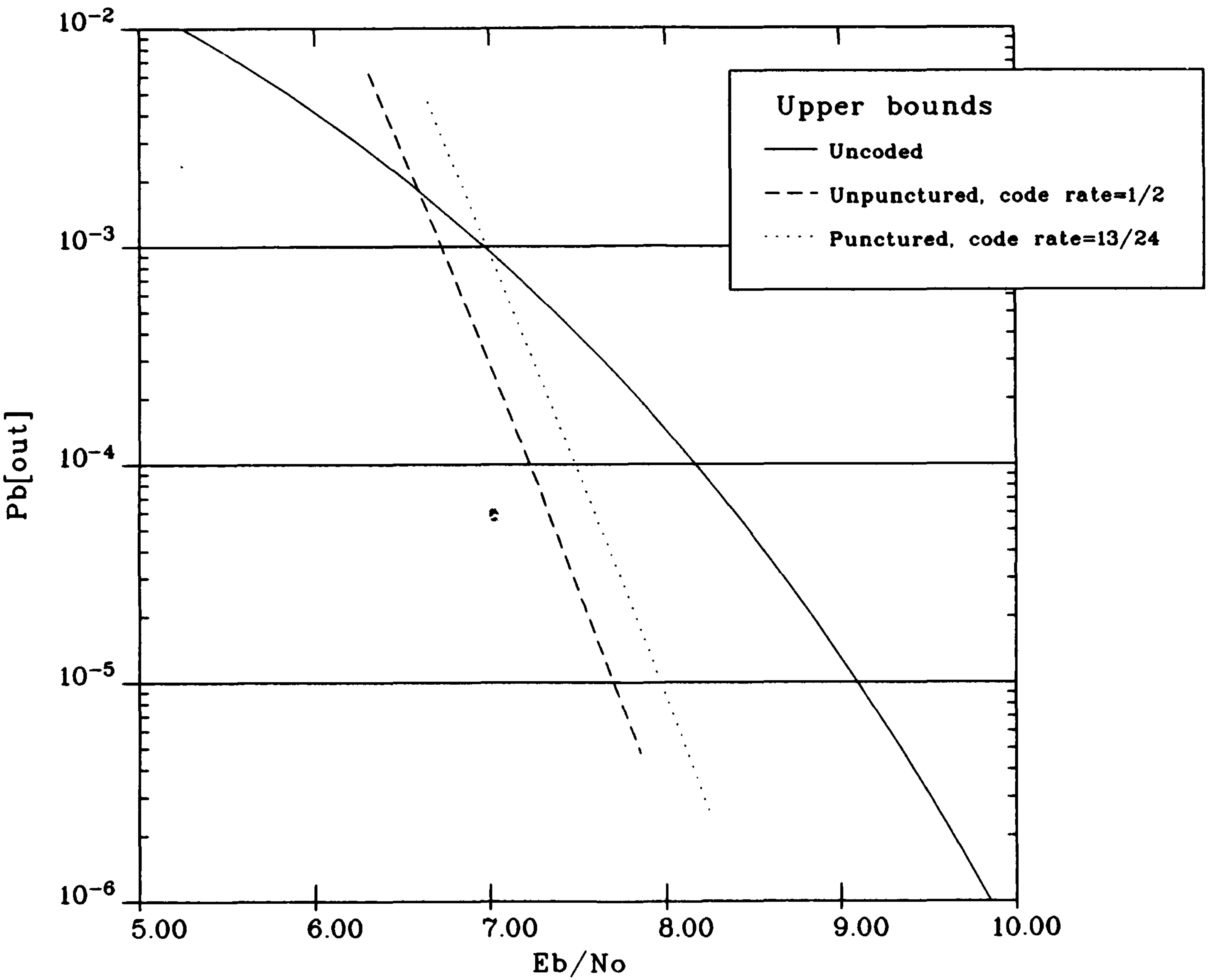


Fig. 4 Comparison of error bounds.

7. IMPROVEMENT OF TIMING ROBUSTNESS

The solution presented in the previous sections treats the control bits as requiring the same protection against jamming and noise as the data. An error in a data bit would not propagate to other bits, but an error in a control bit would cause the timing of all ensuing bits to slide by at least one bit. In an military operational environment this would be a serious error, causing the decryption and decoding to lose synchronisation. The link would have to be re-synchronised.

A method to increase the resilience of the control bit to noise and/or jamming is to add redundant bits. A scheme using a 27/48 RCPC is proposed in Table 2. This scheme increases the Hamming distance (a measure of the error correction capability) between critical control words, thereby decreasing the probability of error. Bit positions 1 to 4 contain the control bits and indicate whether there are 23 or 25 bits of data. If there are 25 bits of data, bit positions 1 to 4 also determine the value of data bits 24 and 25. Bit positions 5 to 27 contain data bits 1 to 23.

<table>
<tr><th colspan="7">Bit Position</th><th colspan="2" rowspan="2">Value of data bits 24 & 25</th></tr>
<tr><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>...</th><th>27</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td colspan="3" rowspan="5">data bits
1 to 23</td><td colspan="2">unused</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr>
</table>

Table 2. Definition of control bits for jammer resistance.

The two critical control words in the proposed scheme are the ones which determine if there is 23 bits of data or 25 bits of data. The Hamming distance between these two control words is 3, indicating that the scheme can correct single bit errors. Therefore, for an timing error to occur, two of the control bits must be in error. This improvement in timing robustness is demonstrated in the comparisons in table 3.

Probability of data error	1.0×10^{-3}	1.0×10^{-4}	1.0×10^{-5}	1.0×10^{-6}
Probability of timing error	6.0×10^{-6}	6.0×10^{-8}	6.0×10^{-10}	6.0×10^{-12}

Table 3. Comparison of probability of error for data and timing

To implement the scheme defined above the number of data bits plus control bits must be increased to 27. In order for the overall coded data rate to remain constant, a rate 27/48 (or 9/16) code will be required. The performance of this code is compared with the previous codes in Fig. 5, showing a degradation of 0.5 dB from the unpunctured code. If the allowable variation in the data rate was reduced to ±2%, a scheme could be implemented with a rate 51/96 (or 17/32) code. The performance of this code is also presented in Fig. 5, showing a degradation of only 0.2 dB from the unpunctured code.

8. OTHER USES OF RCPC

The previous sections have described how a RCPC can be used to implement a concept to support equipment with non-synchronised clocks. This section will describe how a RCPC can be used to transmit data rates which are not exactly

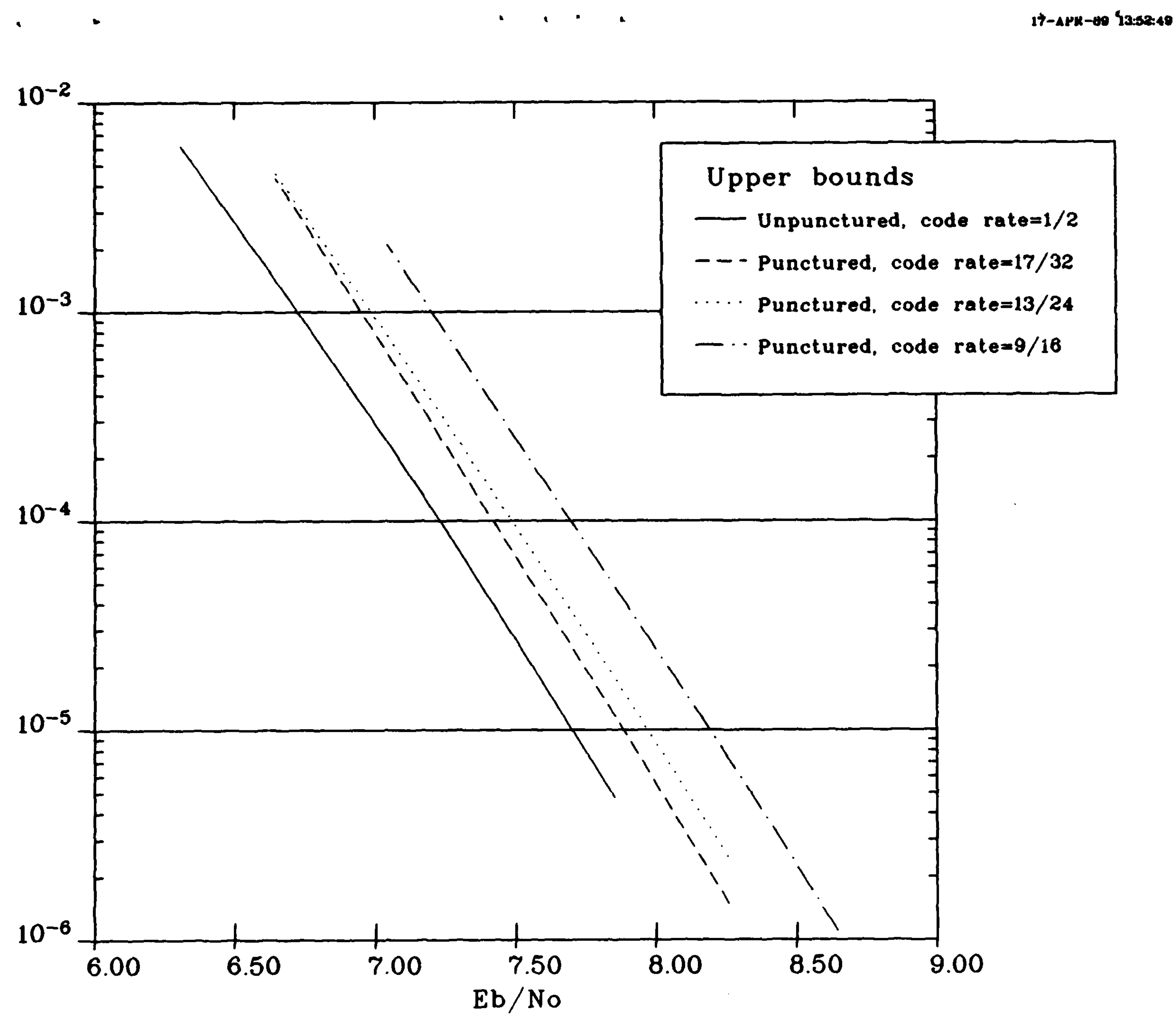


Fig. 5 Comparison of error bounds.

matched to the data rate of the data communication system, without using filler bits.

Many MILSATCOM systems assign communication resources in increments of 1200 bps. Therefore, a 16 kbps digital voice transmission would normally require a 16.8 kbps link, using filler bits. Using a rate 20/39 code, the 16 kbps voice could be transmitted over a nominal 15.6 kbps link, requiring no filler bits and using less bandwidth.

A T1 carrier is defined as 1,536 kbps of data and 8 kbps of framing, for an overall data rate of 1,544 kbps. A typical communication system handling medium data rate information would use transmission rates of 32 kbps, or multiples of 32 kbps. This would imply that a 1,544 kbps data rate would require a 1,568 kbps channel using filler bits. Instead of using this approach a rate 25/48 RCPC would allow the full T1 carrier (1,544 kbps) to be transmitted over the nominal 1,536 kbps link without filler bits.

9. CONCLUSION

The concept described in this document provides a simple method of implementing a scheme to allow a variation of $\pm 4\%$ in the source data rate for a system with a fixed symbol transmission rate. The degradation in performance when compared to the existing scheme, which allows no variation in the transmission rate, is 0.3 dB. This is considered a very acceptable performance sacrifice. In addition, little modification is needed to the data link standard and technical complications are kept to a minimum.

APPENDIX A

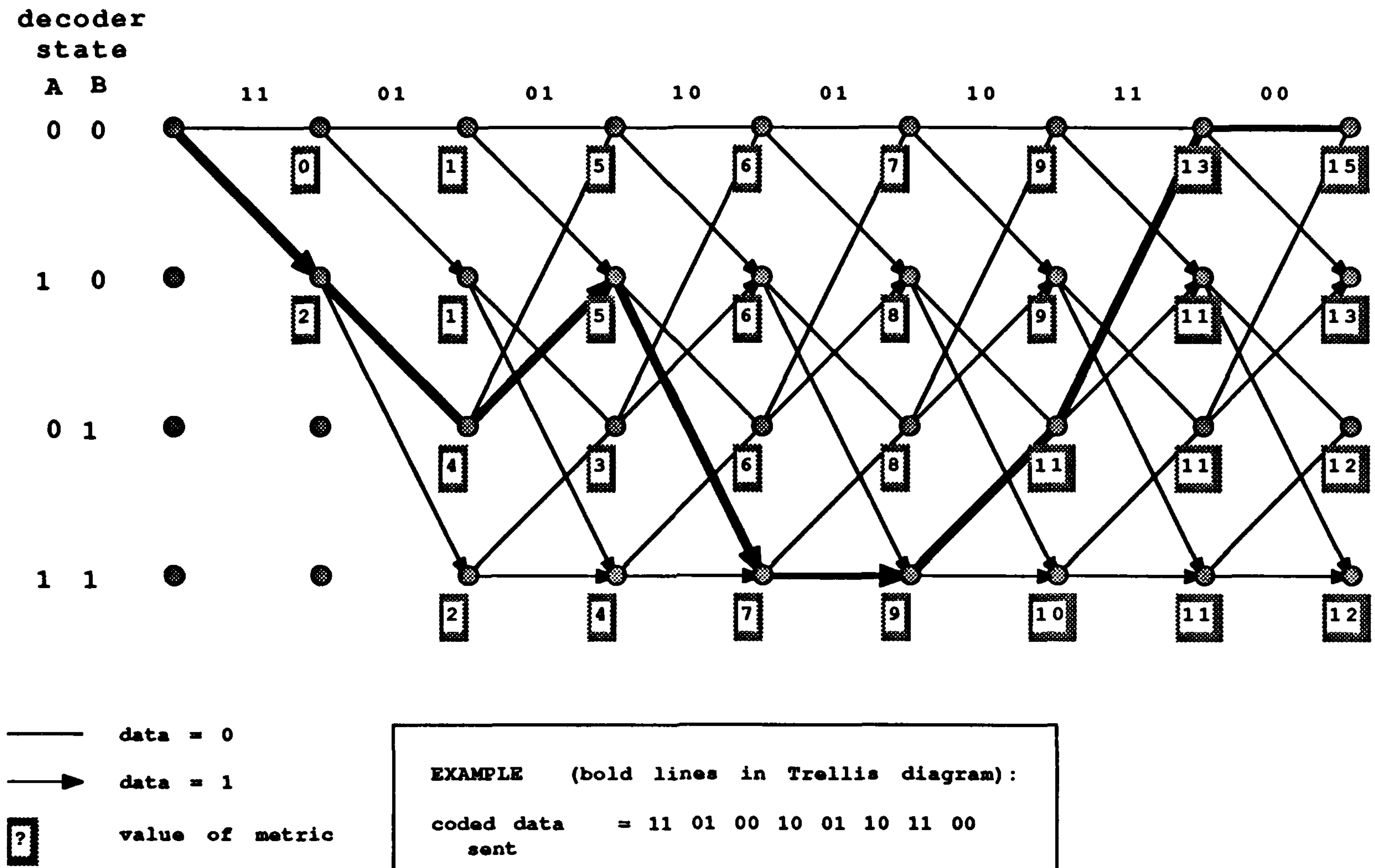
Calculation of the Upper Bound

For Convolutional Codes

This appendix describes a method to determine the performance bounds on specific convolutional codes using binary symmetric channels. The initial work was given in [3] for standard convolutional codes with extensions in [2] for RCPC codes.

The decoding of a convolutional code is usually described as moving through a Trellis diagram, as shown in Fig. A1. If the decoder employs the Viterbi algorithm it is classed as maximum-likelihood decoder in that it selects the path with the maximum likelihood function (or metric) as the correct path in the Trellis. As shown in Fig. A1 the path chosen is the path with the largest metric. The measure of the performance of a code is usually given by the bit error probability, P_b. In convolutional codes, and RCPCs, it is not possible to explicitly define the probability of bit error (P_b), but P_b can be upper bounded.

Before calculating the upper bound on P_b, we will turn to the determination of the more readily available P_n, the error probability per node. Referring to Fig. A2, we see two paths originating at node j. Without loss of generality, we take the all zeros path to be the correct path and the other



EXAMPLE (bold lines in Trellis diagram):

coded data sent = 11 01 00 10 01 10 11 00

coded data received = 11 01 01 10 01 10 11 00

decoded data = 1 0 1 1 1 0 0 0

Fig. A1. Example of decoded data with one coded bit corrupted by noise.

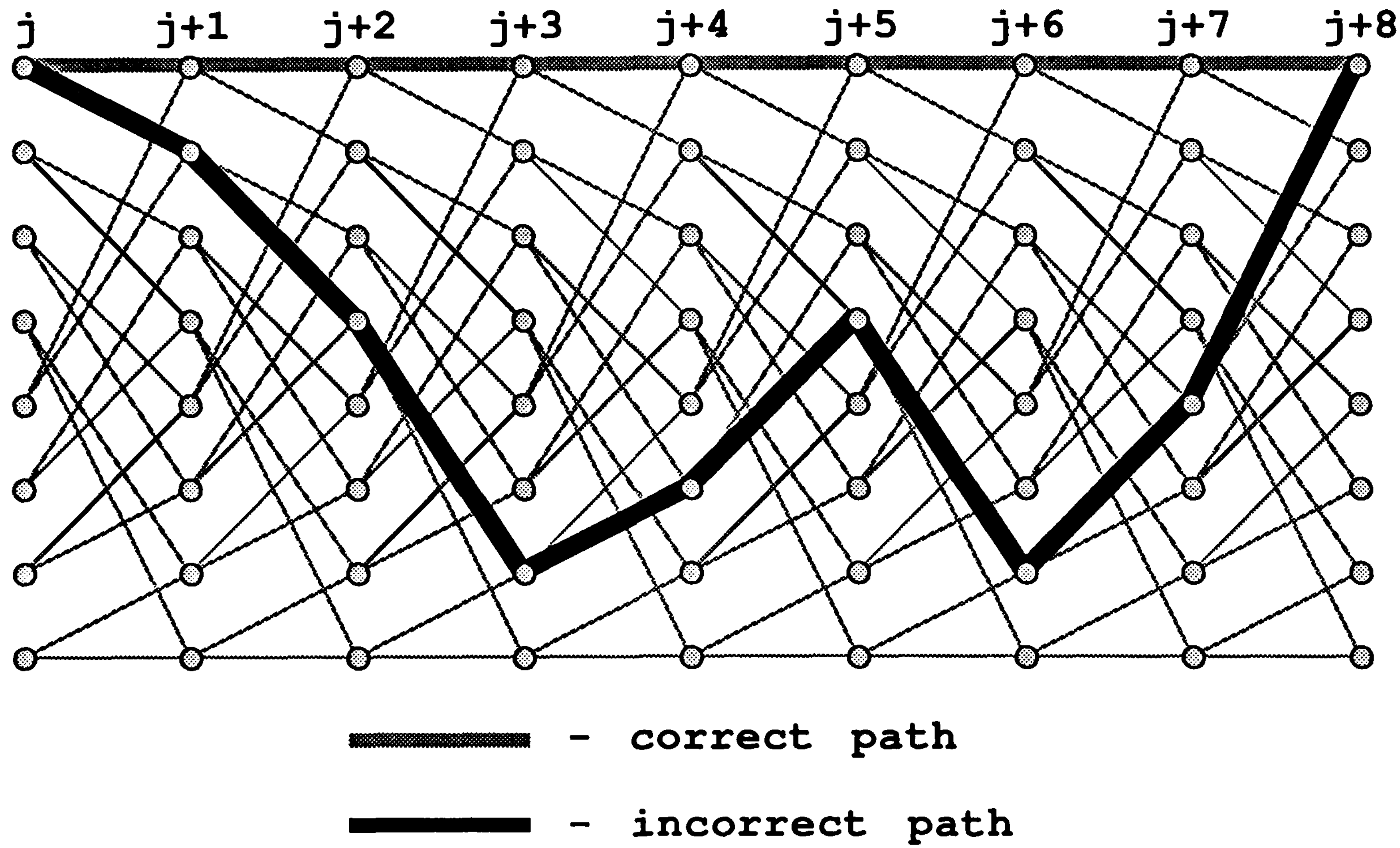


Fig. A2. Example of an incorectly decoded path in a Trellis diagram.

path to be an incorrect one chosen by the decoder. For this error to occur the correct path over the unmerged segments must have a lower metric than the incorrect path. Therefore, we can upper bound P_e (employing the union bound) by

$$P_n(j) \leq \sum_{x_j' \text{ is an element of } \chi'(j)} \Pr [\ \Delta M(x_j', x_j) \geq 0\ ] \qquad \text{...(A1)}$$

where x_j' = is the incorrect path diverging at, node j

$\chi'(j)$ = is the set of all such paths, and

$\Delta M(x_j', x_j)$ = is the metric difference between the correct and incorrect path.

Each term of the summation is the pairwise error probability for the two code vectors over the unmerged segment which yields

$$P_n(j) \leq \sum_{d = d_f}^{\infty} a(d)\ P_d \qquad \text{... (A2)}$$

where d_f = minimum free distance of code,

$a(d)$ = incorrect path's Hamming distance d from the correct path, and

P_d = pairwise error probability for code vectors at distance d.

The value P_b can now be bounded by weighting each term of the union bound by the number of bit errors which occur on that path. This corresponds to the number of "1"s in the data sequence over the unmerged segment of the incorrect path

(the all zeros path was selected as the correct path). Therefore,

$$P_b \leq \sum_{i=1}^{\infty} \sum_{d=d_f}^{\infty} ia(d,i)P_d \qquad \text{... (A3)}$$

where i = number of "1"s in the data segment of the incorrect path.

The exact values of P_d can be determined for a binary symmetric channel (BSC) using

$$P_d = \begin{cases} \displaystyle\sum_{k=(d+1)/2}^{d} \binom{d}{k} p^k (1-p)^{d-k} & d \text{ odd} \\ \displaystyle\frac{1}{2}\binom{d}{d/2} p^{d/2}(1-p)^{d/2} + \sum_{k=(d+1)/2}^{d} \binom{d}{k} p^k (1-p)^{d-k} & d \text{ even} \end{cases} \qquad \text{... (A4)}$$

where p = input error probability.

To use the above method for RCPC codes the time varying nature of the code must be accounted for. P (period of punctuation) different starting points for diverging paths need to be considered. Equation (A3) then becomes

$$P_b \leq \frac{1}{P} \sum_{i=1}^{\infty} \sum_{d=d_f}^{\infty} i a(d,i) P_d \qquad \ldots \text{(A5)}$$

The values a(d,i) and i were determined from a computer program that searched for all possible paths up to length 20. Path lengths greater than 20 had very little effect on the value of the upper bound for input probabilities $\leq 10^{-2}$. This program works with all rate 1/2 convolutional codes up to constraint length 8 or any RCPC derived from these convolutional codes. Using a convolutional code of rate 1/2, constraint length 7, and generator matrix 171_8, 133_8, the upper bound is

$$P_b = 36P_d(10) + 211P_d(12) + 1404P_d(14) + 11633P_d(16) + 77433P_d(18) + 502690P_d(20).$$

This is the same result obtained by Marvin K. Simon et al [5]. The upper bound for the optimised RCPC code using the above convolutional code and puncturing values of 07677_8 and 177777_8 is

$$P_b = 3P_b(8) + 26P_b(9) + 63P_b(10) + 162P_b(11) + 457P_b(12) + 1298P_b(13) + 3908P_b(14) + 9896P_b(15) + 22465P_b(16) + 48068P_b(17) + 95527P_b(18) + 177398P_b(19) + 302119P_b(20).$$

REFERENCES

[1] George C. Clark, Jr. and J. Bibb Cain, "Error-Correction Coding for Digital Communications, Plenum Press, New York, June 1981.

[2] Joachim Hagenauer, "Rate-Compatible Punctured Convolutional Codes (RCPC Codes) and their Applications", IEEE Transactions on COmmunications, Vol. 36, no. 4, April 1988.

[3] Andrew J. Viterbi and Jim K. Omura, "Principles of Digital Communications and Coding", McGraw-Hill, United States, 1979.

[4] John G. Proakis, "Digital Communication", McGraw-Hill, United States 1983.

[5] Marvin K. Simon, Jim K Omura, Robert A. Scholtz, and Barry K. Levitt, "Spread Spectrum Communications", Computer Science Press, United States, 1985.

DOCUMENT CONTROL DATA

(Security classification of title, body of abstract and indexing annotation must be entered when the overall document is classified)

1. ORIGINATOR (the name and address of the organization preparing the document. Organizations for whom the document was prepared, e.g. Establishment sponsoring a contractor's report, or tasking agency, are entered in section 8.) Defence Research Establishment Ottawa Ottawa, Ontario K1A 0Z4	2. SECURITY CLASSIFICATION (overall security classification of the document including special warning terms if applicable) UNCLASSIFED

3. TITLE (the complete document title as indicated on the title page. Its classification should be indicated by the appropriate abbreviation (S,C,R or U) in parentheses after the title.)

Rate Compatible Punctured Codes, Their Use in Military Satellite Communication Systems (U)

4. AUTHORS (Last name, first name, middle initial)

Wagner, Lyle C.

5. DATE OF PUBLICATION (month and year of publication of document) June 1989	6a. NO. OF PAGES (total containing information. Include Annexes, Appendices, etc.) 26	6b. NO. OF REFS (total cited in document) 5

7. DESCRIPTIVE NOTES (the category of the document, e.g. technical report, technical note or memorandum. If appropriate, enter the type of report, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.)

DREO Technical Note

8. SPONSORING ACTIVITY (the name of the department project office or laboratory sponsoring the research and development. Include the address.)

Electronics Division/SATCOM Section
Defence Research Establishment Ottawa
Ottawa, Ontario K1A 0Z4

9a. PROJECT OR GRANT NO. (if appropriate, the applicable research and development project or grant number under which the document was written. Please specify whether project or grant) 041LM	9b. CONTRACT NO. (if appropriate, the applicable number under which the document was written)
10a. ORIGINATOR'S DOCUMENT NUMBER (the official document number by which the document is identified by the originating activity. This number must be unique to this document.) DREO Technical Note 89-19	10b. OTHER DOCUMENT NOS. (Any other numbers which may be assigned this document either by the originator or by the sponsor)

11. DOCUMENT AVAILABILITY (any limitations on further dissemination of the document, other than those imposed by security classification)

(X) Unlimited distribution
( ) Distribution limited to defence departments and defence contractors; further distribution only as approved
( ) Distribution limited to defence departments and Canadian defence contractors; further distribution only as approved
( ) Distribution limited to government departments and agencies; further distribution only as approved
( ) Distribution limited to defence departments; further distribution only as approved
( ) Other (please specify):

12. DOCUMENT ANNOUNCEMENT (any limitation to the bibliographic announcement of this document. This will normally correspond to the Document Availabilty (11). However, where further distribution (beyond the audience specified in 11) is possible, a wider announcement audience may be selected.)

UNCLASSIFIED

SECURITY CLASSIFICATION OF FORM

13. ABSTRACT (a brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), (R), or (U). It is not necessary to include here abstracts in both offical languages unless the text is bilingual).

The problem of interfacing older military terminal equipments, which work exclusively from an internal timing source, into a communication system that requires all terminals to synchronise to the network clock is addressed in this paper. These data communication systems would typically operate with a fixed symbol transmission rate and employ end-to-end convolutional coding for channel errors. A scheme is proposed which would allow a clock discrepancy of up to ±4% between the terminal clock and the network clock. The method proposed is to add control bits to the information stream between terminals. The control bits would allow the effective information rate to be either increased or decreased by ±4%. The overall symbol rate of the communication link is held constant by selecting a higher rate convolutional code from the set of rate compatible punctured codes (RCPCs). This paper describes the methodology used to select an optimal RCPC and presents some examples. The probability of error for a selected code is compared with that of a convolutional code for an application typical of a satellite communication system. The conclusion is that the increase in signal to noise required to sustain the same probability of error is minimal.

14. KEYWORDS, DESCRIPTORS or IDENTIFIERS (technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible keywords should be selected from a published thesaurus. e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus-identified. If it is not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title.)

Rate Compatible Punctured Codes
Military Communication
Satellite Communication
Jamming
Computer Simulation

UNCLASSIFIED

SECURITY CLASSIFICATION OF FORM